# ANCIENT TEXT TRANSLATOR


**By-**

| | |
|---|---|
| **Nitish Pant** | **(1609113070)** |
| **Sanchit Garg** | **(1609113096)** |
| **Syed Md. Farhan** | **(1609113115)** |
| **Tushar Ahuja** | **(1609113117)** |

**Under the supervision of**

**Assistant Professor Mrs. Shikha Verma**

**Department of Information Technology**

**JSS ACADEMY OF TECHNICAL EDUCATION**

**Sector 62, Noida, Uttar Pradesh 201301**


**2019-20**

# DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma from a university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature:
Name : Nitish Pant
Roll No.: 1609113070
Date :

Signature:
Name : Sanchit Garg
Roll No.: 1609113096
Date :

Signature:
Name : Syed Md. Farhan
Roll No.: 1609113115
Date :

Signature:
Name : Tushar Ahuja
Roll No.: 1609113117
Date :

# CERTIFICATE

This is to certify that Project Report entitled "**ANCIENT TEXT TRANSLATOR"** which is submitted by **Nitish Pant, Sanchit Garg, Syed Md. Farhan, Tushar Ahuja** in partial fulfillment of the requirement for the award of degree B.Tech. in Department of Information Technology of **Dr. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY**, is a record of the candidates' own work carried out by them under my supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

**Supervisor: Asst. Prof. Mrs. Shikha Verma**

**Date : 14/06/2020**

# ACKNOWLEDGEMENT

*It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe a special debt of gratitude to Mrs. Shikha Verma, Department of Information Technology, JSSATEN, Noida for his constant support and guidance throughout the course of our work. His sincerity, thoroughness, and perseverance have been a constant source of inspiration for us. It is only his cognizant effort that our endeavors have seen the light of the day.*

*We also take the opportunity to acknowledge the contribution of Dr. Vineeta Khemchandani, Head, Department of Information Technology Engineering, JSSATE, Noida for her full support and assistance during the development of the project.*

*We also do not wish to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not least, we acknowledge our friends for their contribution in the completion of the project.*

*Signature:*
*Name :* *Nitish Pant*
*Roll No.:* *1609113070*
*Date :*

*Signature:*
*Name :* *Sanchit Garg*
*Roll No.:* *1609113096*
*Date :*

*Signature:*
*Name :* *Syed Md. Farhan*
*Roll No.:* *1609113115*
*Date :*

*Signature:*
*Name :* *Tushar Ahuja*
*Roll No.:* *1609113117*
*Date :*

# ABSTRACT

Ancient Text Translator software today provides adequate conversion of ancient languages to one's native tongue or mediator language like English; however, dialects, slang, and character conversion errors result in partially successful translations. For an accurate translation, a native speaker is often required to correct the translation by using sentence structure and word use cues to capture the true meaning. Text Translator character conversion from Sundanese languages to western characters induce errors in the translated text which can change the meaning or result in characters being associated together that do not form words. The authors present a solution using Machine Translation. The solution provides proper character conversion to achieve greater translation accuracy. This Ancient Text Translator can be helpful in many ways like when we visit any monument and encounter any guide stone or memorial stone having engraved text of ancient language or while we're reading any piece of work of that language. Language translation has completely changed the pace of marketing worldwide. Previously, the language was the biggest hurdle in conversing, but this problem has been slowly disappearing as language translator applications become more effective. The Source Language for this Project is 'Sundanese' and the target language is 'English'.

# TABLE OF CONTENTS

# LIST OF TABLES

**CONTENT** **PAGE NO.**

# LIST OF FIGURES

**CONTENT** **PAGE NO.**

# LIST OF ABBREVIATIONS

| | |
|---|---|
| ML | Machine Learning |
| ANN | Artificial Neural Networks |
| RNN | Recurrent Neural Networks |
| LSTM | Long Short term memory |
| GRU | Gated Recurrent Units |
| CNN | Convolutional Neural Networks |
| POS | Part of Speech |
| API | Application Programming Interface |
| REST | Representational state transfer |
| AI | Artificial Intelligence |

# OBJECTIVE

Language is the only way to communicate with people from different regions. To be able to clearly convey our message to another person we need to have a good hold of their language, in the modern days, this barrier of language can be reduced or even eliminated by using a text translator.

This Ancient Text Translator can be helpful in the following conditions-

1-To collect and analyze the ancient script images available in the palm leaf literature format.

2- To pre-process i.e by removing the stop word, various punctuations and finally converting all the upper case into lower case,

3-To translate the recognizable script using vectorization and Long Short Term Memory (LSTM) modeling into a current recognizable format i.e English.

4 To verify and validate (V&V) the developed ancient script text translation system by performing comparative analysis with the various existing systems.

1- When we visit any monument and encounter any guide stone or memorial stone having engraved text of ancient language.

2- When we're reading any piece of work of that language.

The main aim is to translate the given ancient language to the target language i.e 'English',

For this, we would require a text mapping from that language to the target language.

# MOTIVATION

The main motivation is derived from the fact When we visit any monument and encounter any guide stone or memorial stone having engraved text of ancient language, we aren't able to understand that language.

So, it is important to get the knowledge of our traditions that used to happen in the past and the morals which were implemented by our ancestors. To implement and understand our ancestor's sayings we need to decode or decipher ancient writings into a common medium and that would be English.

Today there are more than 6800 languages spoken globally, and in this increasingly globalized world, every culture has interaction with every other culture in some.

That means there is an incalculable number of translations required every second of every day, which has never been an easy task to do.

Whether an idea or story or a quest, at least one message is lost in translation.

# CHAPTER 1
# INTRODUCTION

As we all know that Google Translate, a website that has a variety of languages that can translate into another language instantly. It has made the world a nice place by increasing communication with different background people when language is a barrier to them.

It not only changed the level of communication but it also increased the business a lot( through advertising). Previously, the language was the biggest hurdle in conversation, but this problem has been slowly disappearing as language translator applications become more effective.

But even Google Translate can't translate the ancient language whose data is not present in an organized way. Our language translator will interpret the structure of the sentence in the source language and will generate a translation of it into the language the user is translating to.

The Source Language for this Project is 'Sundanese' and the target language is 'English'.

We will be using sequence-to-sequence learning, it is a very powerful technique that is used to solve many kinds of problems. After that, we will use neural machine translation for further translation.

## Machine Learning Methods Used

ML is undoubtedly one of the most powerful technologies that this world has ever seen. It is a subpart of Artificial Intelligence, that develops an ability to systems or programs to learn automatically i.e. from their past experience which helps them in enhancing their performance. It focuses on the development of computer programs that access data and specific techniques to get the desired results. ML techniques are used to find the learning patterns within complex data that we would otherwise struggle to discover. Then, the hidden patterns and feedback is used to predict the desired result.

Following are the used ML algorithms:

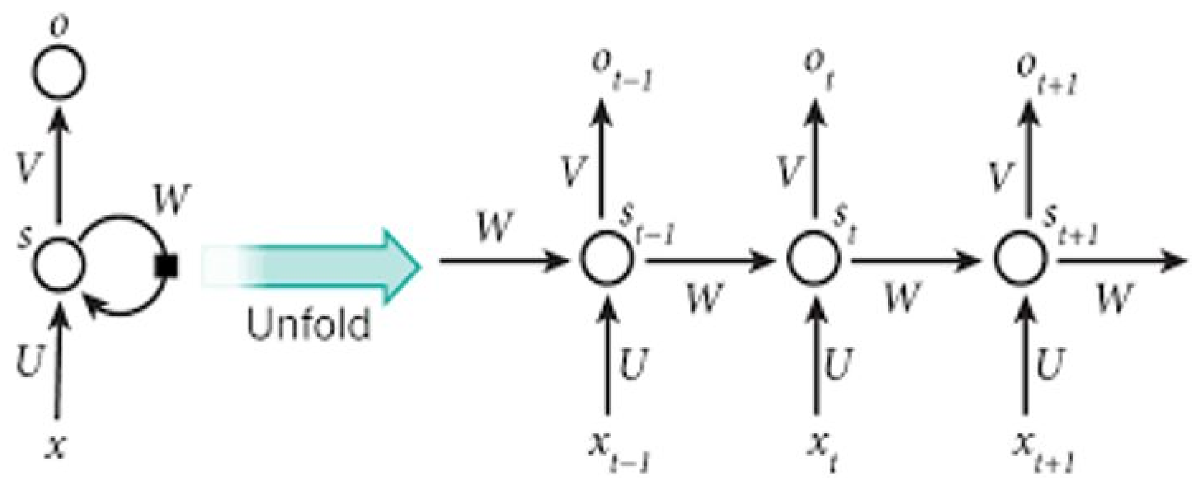
## Recurrent neural networks-

A recurrent neural network is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behavior. Unlike feedforward neural networks, RNNs can use their internal state to process sequences of inputs.

Here there is a simple multiplication of Input (xt) and Previous Output (ht-1). Passed through the Tanh activation function. No Gates present.

During backpropagation, recurrent neural networks suffer from the vanishing gradient problem.

Gradients are values used to update neural network weights. The vanishing gradient problem is when the gradient shrinks as it back propagates through time. If a gradient value becomes extremely small, it doesn't contribute too much learning.




## NLP

Natural language processing is a subfield of linguistics, computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human languages, in particular how to program computers to process and analyze large amounts of natural language data.

## Pytorch

PyTorch is an open-source machine learning library based on the Torch library, used for applications such as computer vision and natural language processing. It is primarily developed by Facebook's AI Research lab. It is free and open-source software released under the Modified BSD license.

## LSTM-Long short-term memory (LSTM)

It is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points (such as images), but also entire sequences of data

## GRU - Gated Recurrent Units

A GRU is an RNN where an Update gate is introduced, to decide whether to pass Previous O/P (ht-1)to next Cell (as ht) or not. Forget gate is nothing but additional Mathematical Operations with a new set of Weights (Wt).

## POS tagging

POS tagging is a technique that is used to mark up a word in a corpus to a particular part of a speech tag which is entirely based on the context. This is not a straight forward task as it seems, because a particular word may have a different part of speech based on the context in which the word is used.

# CHAPTER 2

# SUMMARY OF PREVIOUS RELATED WORK

| Year | Author | Approach | Dataset | Accuracy | Pros/Cons |
|---|---|---|---|---|---|
| 2019 | Mozhdeh Gheini, Jonathan May | Word segmentation algorithm, sequence to sequence mapping, LSTM. | 9 different language pairs | 86.90 % | + Worked on RGB videos with high accuracy<br>+ 3 variables are used making it a 3-way process where 2nd variable once trained and produce ego masks for all input features<br>- Training time is very high |
| 2017 | Adam Geitgey | Language Translation with Deep Learning and the Magic of Sequences | Multi-lingual word corpus | 75.42 % | + Training was fast<br>+ Encoder network was small and simple<br>- Weak encoder gave low accuracy |

| | | | | | |
|---|---|---|---|---|---|
| 2015 | Thomas Tracey | Language Translation with RNNs | Linguistic word corpus | 74.77 % | + Training was faster<br>+ The pre-trained network was used<br>- The output of encoder i.e VGG16 is weak for classification |
| 2013 | Shashi Pal Singh, Ajai Kumar | MACHINE TRANSLATION USING DEEP LEARNING: AN OVERVIEW | English - Hindi word corpus | 84.2% | + A light-weight detector is used to detect the sequence<br>+ Large dataset for training |
| 2016 | Dzmitry Bahdanau, Kyung Hyun Cho, Yoshua Bengio | NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE | English - German, word corpus | 78.3% | + Better than the conventional model(RNNencdec) at translating long sentences.<br>+ Monotonic alignment helps model correctly translate a phrase. |

| | | | | | |
|---|---|---|---|---|---|
| 2018 | Yong Cheng , Zhaopeng Tu , Fandong Meng , Junjie Zhai and Yang Liu | Robust Neural Machine Translation using Adversarial Learning | Chinese-English, English-German and English-French | 68% | + Data augmentation improve the robustness of NMT models.<br>+ Adversarial stability training aims to stabilize both the encoder and decoder in NMT models. |
| 2017 | Mikel Artetxe, Gorka Labaka, Eneko Agirre, Kyunghyun Cho | UNSUPERVISED NEURAL MACHINE TRANSLATION USING DUAL DECODER | French-English | 84.3% | - Have difficulties to preserve some concrete details from source sentences.<br>+ Able to produce high-quality translations. |
| 2016 | Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, Koray Kavukcuoglu | NEURAL MACHINE TRANSLATION IN LINEAR TIME(DYNAMIC UNFOLDING) | English-German | 81% | + It has short signal propagation paths for tokens in sequences.<br>+ Uses Character level machine translation.<br>- Character Prediction is not very Accurate |

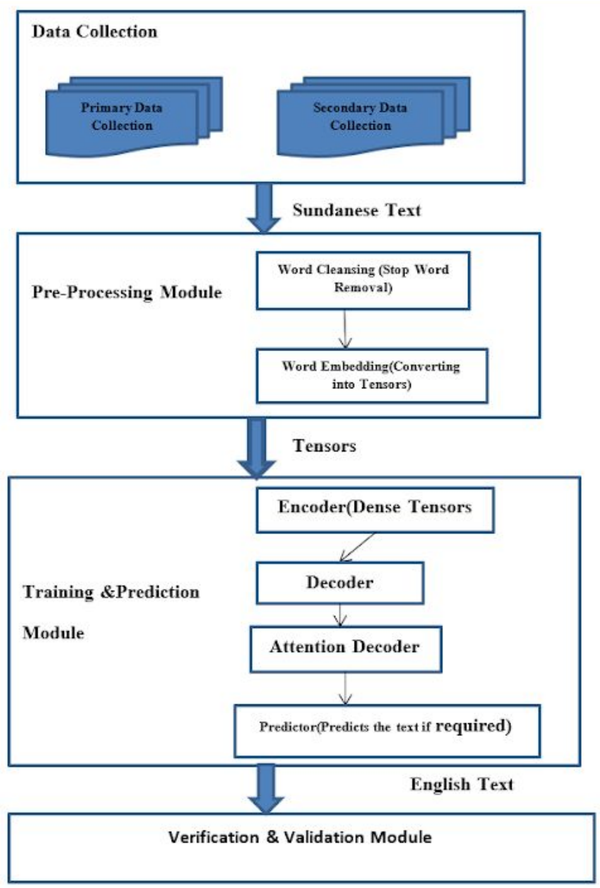
| | | | | | |
|---|---|---|---|---|---|
| 2017 | Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato | UNSUPERVISED MACHINE TRANSLATION USING MONOLINGUAL CORPORA | English-French, English-German | 77.6% | + Subsequent iterations yield significant gains although with diminishing returns.<br>- Simply removes unknown words |

Table 1: Summary of previous work done on Language translation

# CHAPTER 3

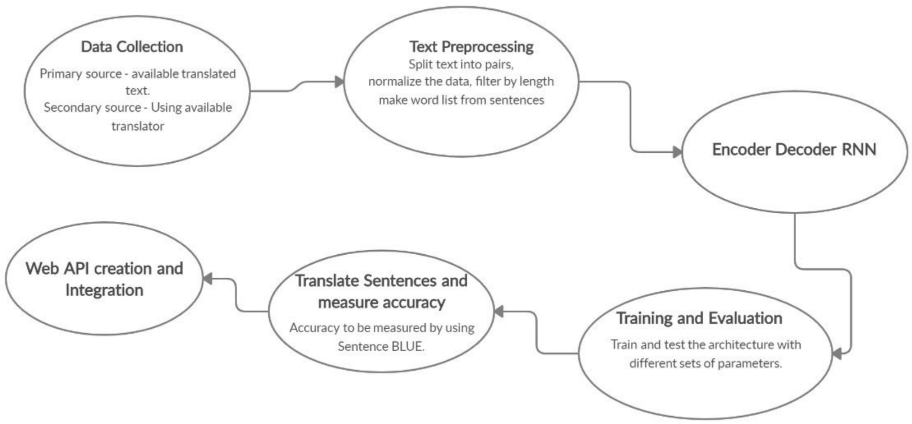
## SYSTEM DESIGN AND ARCHITECTURE

### 3.1 Architecture




The Architecture of the model is based on the Sequence to Sequence model which is also known as the Encoder-Decoder model, the input text is firstly converted into text and is then fed into the translation model which is trained and evaluated.

## 3.3 Data Flow Diagram




## 3.4 Proposed Methodology

Input data will include text in some ancient language and the task will be to convert the input text into the target language which is 'English'.

The training dataset will include two text files
1- The text file in an ancient language.
2- The English translation of that text file.
The testing dataset will include only the text in the ancient language.
We will reach our goal by following these steps:

### 3.4.1. Clean Text

First, we must load the data in a way that preserves the Unicode Sundanese characters. This would include the following steps.

- Remove all non-printable characters.
- Remove all punctuation characters.
- Normalize the case to lowercase.

- Remove any remaining tokens that are not alphabetic.

### 3.4.2. Split Text

Although we have a good dataset for modeling translation, we will simplify the problem slightly to dramatically reduce the size of the model required, and in turn, the training time required to fit the model.

We will split the text into various different batches before starting with the actual training of the data. We will now split the data into train and test set for model training and evaluation.

### 3.4.3. Data Preprocessing

Since there is a lot of example sentences and we want to train something quickly, we'll trim the data set to only relatively short and simple sentences. Here the maximum length is 10 words.
The full process for preparing the data is:

- Read text file and split into lines, split lines into pairs
- Normalize text, filter by length and content
- Make word lists from sentences in pairs

### 3.4.4. Model Building

We will encode Unicode as the input sequences and English sentences as the target sequences. This is applied to both the training dataset and the test dataset.

It is started by defining the model architecture for further process:

- We will use a GRUlayer and an embedding layer on the encoder side.
- We will use another GRUlayer and followed by a dense layer on the decoder side.

For training of the model, 80% of the data will be used and the left 20% of the data will be used for evaluation.

Finally, we can load the saved model and make predictions on the unseen data. These predictions are sequences of integers. Convert predictions into text (English).

### 3.4.5. The Encoder

The encoder of a seq2seq network is an RNN that outputs some value for every word from the input sentence. For every input word, the encoder builds a vector of words and it contains a hidden state and that hidden state is used to predict the next input word.

Encoder hidden state $h_{ti} = f(h_{(t-1)i}, y_{t-1}, c)$

### 3.4.6.The Decoder

The decoder is basically an RNN that feeds with the encoder output vector and outputs a vector of a sequence of words to create the translation.

**3.4.7. Testing and Deployment:** Testing will be done on a dataset that would be completely unseen for the model.

### 3.4.8 API Design

Flask is a web framework for Python, meaning that it provides functionality for building web applications, including managing HTTP requests and rendering templates. In this project, we have created a basic Flask application that hosts our Deep Learning Model locally.

Flask maps HTTP requests to Python functions. In this API, we've mapped two different URL path ('/') for our initial and final webpages. We've rendered two HTML templates in our API. When we run our Python code, it connects to the Flask server at 'http://localhost:5000/'. Flask then checks if there is a match between the path provided and the defined function and shows us our HTML markup.

In our project, the first HTML markup is a home page welcoming visitors to the site hosting our API. It contains a dropdown menu to select the desired languages, a text area to enter your sentence, and a convert button. Hitting this convert button will make Flask run our Deep Learning Model, redirecting it to our output webpage.

THE second HTML markup is a webpage that displays the result returned from our Deep Learning Model. This page ends with a thanking note to the user.

# CHAPTER 4

# IMPLEMENTATION AND RESULTS

### 4.1 Software and Hardware Requirements

We require optimum feasibility for :

1. Preprocessing the bulk of input data, presently being in the form of texts.
2. Storing the trained deep learning model and
3. For Model Deployment.

To support this ideal requirement we need:

1. Graphical Processing Units specifically (ideally GTX1050 Ti), for training the deep learning model.
2. RAM is also required during the computation ideally an 8 GB of RAM.
3. Libraries to support the model like TensorFlow, Py torch,

With the above-mentioned requirement, the data training and deployment would be smooth and fast. Though the model could be trained on lower specifics of the computers we need to use batch processing which will have slower computation.

Therefore minimum requirements are:

1. 8 GB RAM
2. No dedicated Graphic Card

## 4.2 Snapshot of Interface

### 4.3 Test cases

A total of 300 Sentences were taken for the testing of the model. Some of them are listed below-

- ningali anjeun waktos salajengna
- kunaon kunaon urang persia kuno
- pangéran sanés émut guruna
- tapi kuring hoyong janten pejuang
- tugas bumi sapertos cuci

### 4.4 Results

The Result of the Translation of Sundanese to English of the following text is given below in the table.

| 1 | Sundanese | English Translation |
|---|---|---|
| 2 | kuring langkung resep cicing di bumi tinimbang kaluar | i'd rather rather at home go out of milk |
| 3 | abdi kedah ngadamel keluhan ngeunaan | i have to make a complaint about |
| 4 | murid diajar tina guru sareng ogé | the students from from and and guru |
| 5 | sareng masihan naon anu kénca ka beurit sakedik | and offered what was left to the little mouse |
| 6 | ningali anjeun waktos salajengna | see you next next |
| 7 | kunaon kunaon urang persia kuno | why did the old old |
| 8 | pangéran sanés émut guruna | the other princes remembered their master |
| 9 | tapi kuring hoyong janten pejuang | but i want to be a |
| 10 | tugas bumi sapertos cuci | mundane chores such as |
| 11 | | |

The Translator was trained for a dataset of 6621 pairs of Sundanese- English Sentences, and the model was trained for 45000 epochs, then the encoder and decoder model was saved. Given below is a Snapshot of the training curve.

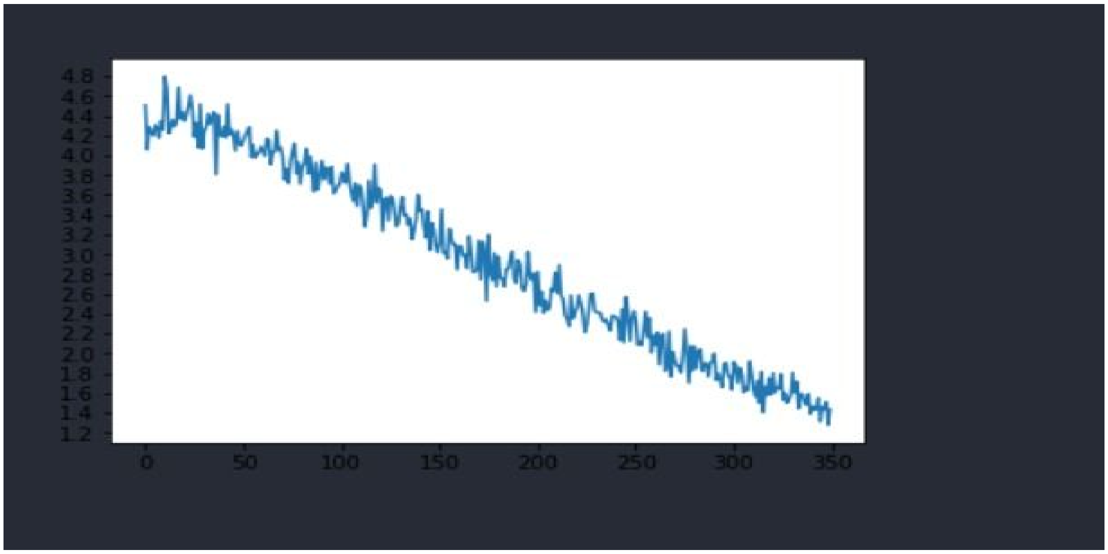
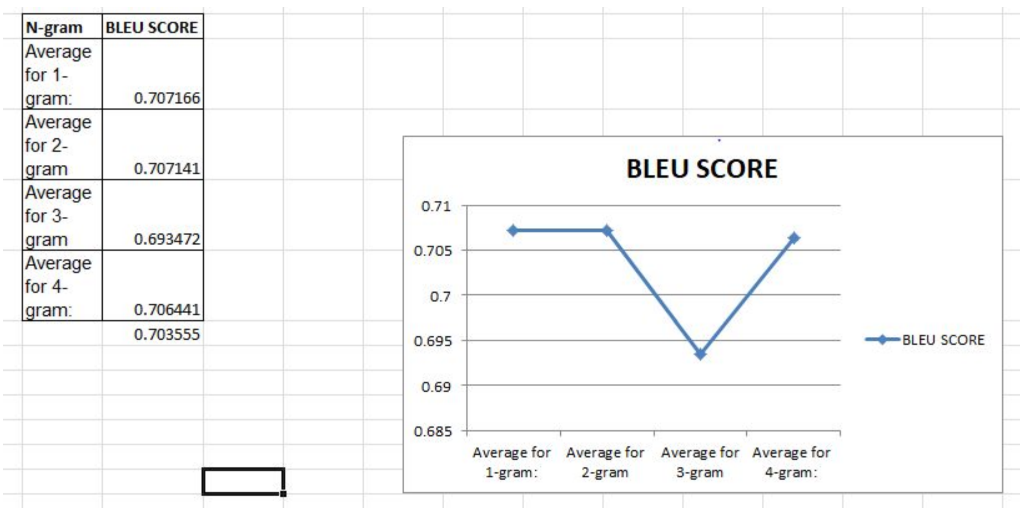



Fig 5. During the training of the model, as time increase the training loss decreases



| N-gram | BLEU SCORE |
|---|---|
| Average for 1-gram: | 0.707166 |
| Average for 2-gram | 0.707141 |
| Average for 3-gram | 0.693472 |
| Average for 4-gram: | 0.706441 |
| | 0.703555 |



Fig 6. Bilingual Evaluation Understudy **Score**. The Bilingual Evaluation Understudy **Score**, or **BLEU** for short, is a metric for evaluating a generated sentence to a reference sentence.

# CHAPTER 5

## CONCLUSION

### 5.1 Performance Evaluation

The performance of our Ancient Text translator which translates Sundanese Language to the English Language was measured using two techniques

(i) Sentence BLEU score

We first compute the n-gram matches sentence by sentence. Next, we add the clipped n-gram counts for all the candidate sentences and divide by the number of candidate n-grams in the test corpus to compute a modified precision score, pn, for the entire test corpus.

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}.$$

(ii) Word Error Rate

The Average Sentence BLEU score of our translator (Sun Trans ) was 71%, and the average word error rate during the testing of the 300 tested sentences was 23%.

This was a remarkable achievement as it is an ancient text translator which is working on a low resource language.

### 5.2 Comparison with existing State-of-the-art technologies

On Comparing with the existing state of the art Sundanese to English translator which is "imtranslator.net ", it was found that our translator gave better results with the sentences involving nouns in them, The overall BLEU score was slightly less than the BLEU score of imtranslator.net but surprisingly the word error rate was better.

Word error rate of imtranslator.net = 0.25

Word error rate of our translator (Sun Trans ) = 0.23

Sentence Bleu score of imtranslator.net = 0.75

Sentence Bleu score of our translator (Sun Trans ) = 0.71

| Sentences | Win Trans | Sun Trans |
|---|---|---|
| Sent 1 | 2 | 1 |
| Sent 2 | 1 | 2 |
| Sent 3 | 0 | 0 |
| Sent 4 | 1 | 1 |
| Sent 5 | 2 | 1 |
| Sent 6 | 2 | 1 |
| Sent 7 | 1 | 1 |
| Sent 8 | 2 | 2 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Win Trans | 2 | 1 | 0 | 1 | 2 | 2 |
| Sun Trans | 1 | 2 | 0 | 1 | 1 | 1 |

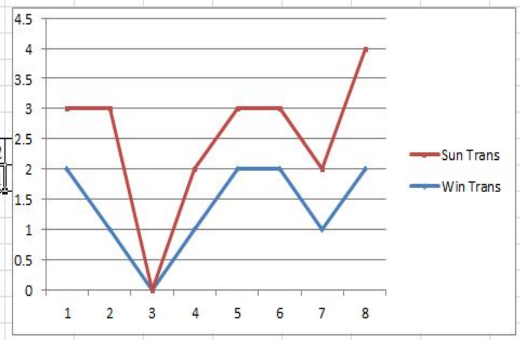



Fig 7. showing word loss graphs for Sun Trans and Win Trans( imtranslator.net )

This is a remarkable achievement as the initial dataset that we have used for training was very less as compared to the dataset which is used by traditional translators.

### 5.3 Future Directions

There is a lot of scope of improvement in this translation, the first improvement that we can do is by increasing the dataset, as it is a low resource language, the most difficult part was creating the dataset, If anyhow we can increase the initial dataset for this translator the accuracy would surely increase.

We can test and train this model with a different number of input layers for the Encoder and decoder.

As the dataset would increase we will also need to change the input parameters for the translator.

Lastly, as the translator works on a general model, we can always include new languages into the translation model just by adding the paired dataset of that language, The ultimate goal would be to mold this translator in such a way that it becomes a multi-lingual translator.

# REFERENCES


1. Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks, 5(2):157-166.
2. Cleeremans, A., Servan-Schreiber, D., and McClelland, J. L. (1989). Finite-state automata and simple recurrent networks. Neural Computation, 1:372{381.
3. Cummins, F., Gers, F., and Schmidhuber, J. (1999). Automatic discrimination among languages based on prosody alone. Technical Report IDSIA-03-99, IDSIA, Lugano, CH.
4. Doya, K. and Shuji, Y. (1989). Adaptive neural oscillator using continuous-time backpropagation learning. Neural Networks, 2(5):375{385. Elman, J. L. (1990). Finding structure in time. Cognitive Science, 14(2):179{211.
5. Fahlman, S. E. (1991). The recurrent cascade-correlation learning algorithm. In Lippmann, R. P., Moody, J. E., and Touretzky, D. S., editors, NIPS 3, pages 190{196.
6. San Mateo, CA: Morgan Kaufmann. Ha
ner, P. and Waibel, A. (1992). Multi-state time delay networks for continuous speech recognition. In Moody, J. E., Hanson, S. J., and Lippmann, R. P., editors, Advances in Neural Information Processing Systems, volume 4, pages 135{142.
7. Morgan Kaufmann Publishers, Inc. Hinton, G. (1986). Learning distributed representations of concepts. In Proceedings of the Eighth Annual Conference of the Cognitive Science Society, pages 1{12, Amherst 1986. Lawrence Erlbaum, Hillsdale.
8. Hochreiter, S. (1991). Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut fur Informatik, Lehrstuhl Prof. Brauer, Technische Universitat Munchen. See www7.informatik.tumuenchen.de/~hochreit.
9. Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8):1735{ 1780.
10. Jordan, M. I. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. In Proceedings of the Eighth Annual Cognitive Science Society Conference, Hillsdale, NJ. Erlbaum.
11. Lin, T., Horne, B. G., Ti~no, P., and Giles, C. L. (1996). Learning long-term dependencies in NARX recurrent neural networks. IEEE Transactions on Neural Networks, 7(6):1329{1338.
12. Mozer, M. C. (1989). A focused backpropagation algorithm for temporal pattern processing. Complex Systems, 3:349{381.
13. Mozer, M. C. (1993). Neural net architectures for temporal sequences processing. In Weigend, A. S. and Gershenfeld, N. A., editors, Time series prediction: Forecasting the future and understanding the past, volume 15, pages 243{264. Addison Wesley, Reading, MA.

14. Mozer, M. C. and Soukup, T. (1991). Connectionist music composition based on melodic and stylistic constraints. In Lippmann, R. P., Moody, J. E., and Touretzky, D. S., editors, Advances in Neural Information Processing Systems, volume 3, pages 789{796. Morgan Kaufmann Publishers, Inc.
15. O'Reilly, R. C., Braver, T. S., and Cohen, J. D. (1999). A biologically-based computational model of working memory. In Miyake, A. and Shah, P., editors, Models of Working Memory: Mechanisms of Active Maintenance and Executive Control. in press.
16. Axelrod, Amittai. "Factored Language Model for Statistical Machine Translation." MRes Thesis. Edinburgh University, 2006.
17. Bertoldi, Nicola, and Marcello Federico. "A New Decoder for Spoken Language Translation Based on Confusion Networks." Automatic Speech Recognition and Understanding Workshop (ASRU), 2005.
18. Bilmes, Jeff A, and Katrin Kirchhoff. "Factored Language Models and Generalized Parallel Backoff." HLT/NAACL, 2003. Koehn, Philipp. "Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models." AMTA, 2004.
19. Koehn, Philipp, Marcello Federico, Wade Shen, Nicola Bertoldi, Ondrej Bojar, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Richard Zens, Alexandra Constantin, Christine Corbett Moran, and Evan Herbst. "Open Source Toolkit for Statistical Machine Translation". Report of the 2006 Summer Workshop at Johns Hopkins University, 2006.
20. Koehn, Philipp, and Hieu Hoang. "Factored Translation Models." EMNLP, 2007.
21. Koehn, Philipp, Franz Josef Och, and Daniel Marcu. "Statistical Phrase-Based Translation." HLT/NAACL, 2003.
22. Och, Franz Josef. "Minimum Error Rate Training for Statistical Machine Translation." ACL, 2003.
23. Och, Franz Josef, and Hermann Ney. "A Systematic Comparison of Various Statistical Alignment Models." Computational Linguistics 29.1 (2003): 19-51.
24. Papineni, Kishore, Salim Roukos, Todd Ward, and WeiJing Zhu. "BLEU: A Method for Automatic Evaluation of Machine Translation." ACL, 2002.
25. Shen, Wade, Richard Zens, Nicola Bertoldi, and Marcello Federico. "The JHU Workshop 2006 Iwslt System." International Workshop on Spoken Language Translation, 2006.
26. Stolcke, Andreas. "SRILM an Extensible Language Modeling Toolkit." Intl. Conf. on Spoken Language Processing, 2002.
27. Zens, Richard, and Hermann Ney. "Efficient PhraseTable Representation for Machine Translation with Applications to Online MT and Speech Recognition." HLT/NAACL, 2007.
28. Daniel Schlegel, “Deep Machine Learning on GPU”, University of Heidelberg-Ziti, 12 January 2015.

29. Holger Schwenk, Yoshua Bengio," Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation", Arxiv: V3,3 Sep 2014.
30. Hugo Larochelle, Yoshua Bengio, Jerome Louradour, Pascal Lamblin, "Exploring Strategies for Training Deep Neural Networks", Journal of Machine Learning Research 1 2009 Submitted 12/07.
31. Ilya Sutskever, Oriol Vinyals, Quoc V. Le, "Sequence to Sequence Learning with Neural Networks", Google.
32. Jiajun Zhang And Chengqing Zong, "Deep Neural Network in Machine Translation", Institute of Automation, Chinese Academy of Sciences.
33. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. (2002)
34. Hopkins, M., May, J.: Tuning as ranking. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK., Association for Computational Linguistics (2011).
35. Bogdan Babych and Anthony Hartley. 2004. Extending the Bleu MT evaluation method with frequency weightings. In Proceedings of ACL.
36. Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for MT evaluation with improved correlation with human judgments. In Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization, Ann Arbor, Michigan.
37. Chris Callison-Burch. 2005. Linear B system description for the 2005 NIST MT evaluation exercise. In Proceedings of the NIST 2005 Machine Translation Evaluation Workshop.
38. Marine Carpuat and Dekai Wu. 2005. Word sense disambiguation vs. statistical machine translation. In Proceedings of ACL.
39. Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for machine translation. In Proceedings of MT Summit IX.
40. Deborah Coughlin. 2003. Correlating automated and human assessments of machine translation quality. In Proceedings of MT Summit IX.
41. George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In Human Language Technology: Notebook Proceedings, pages 128–132, San Diego.